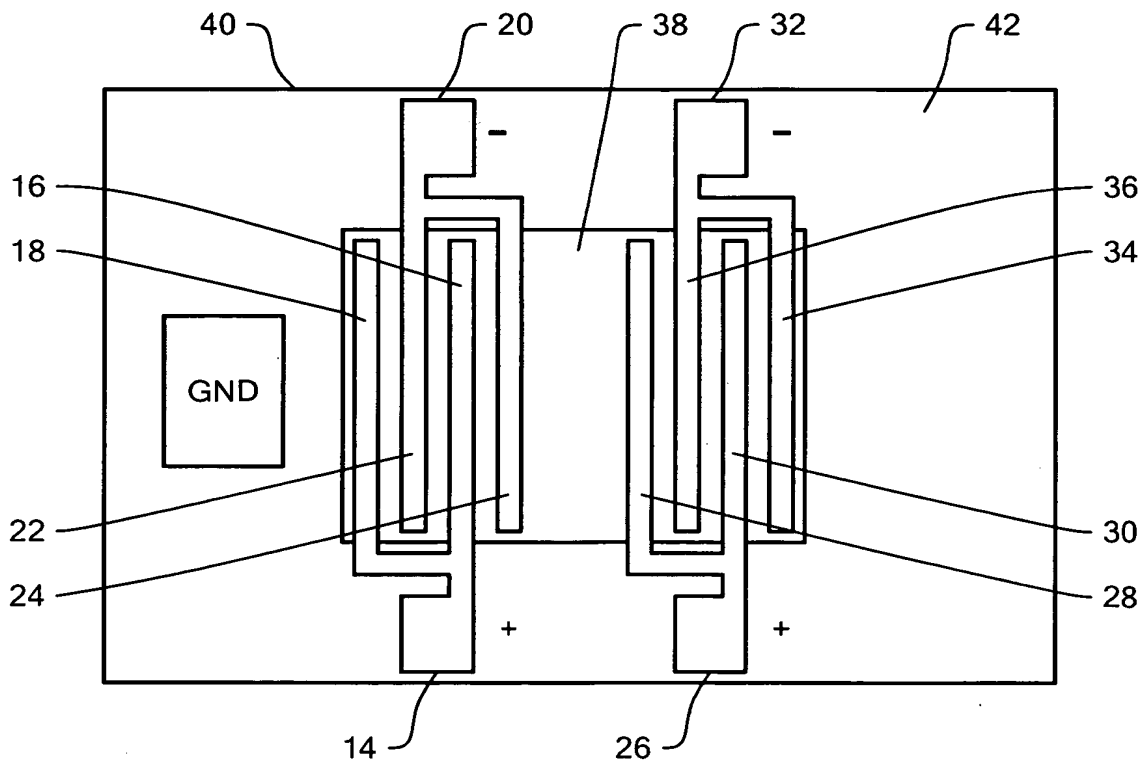


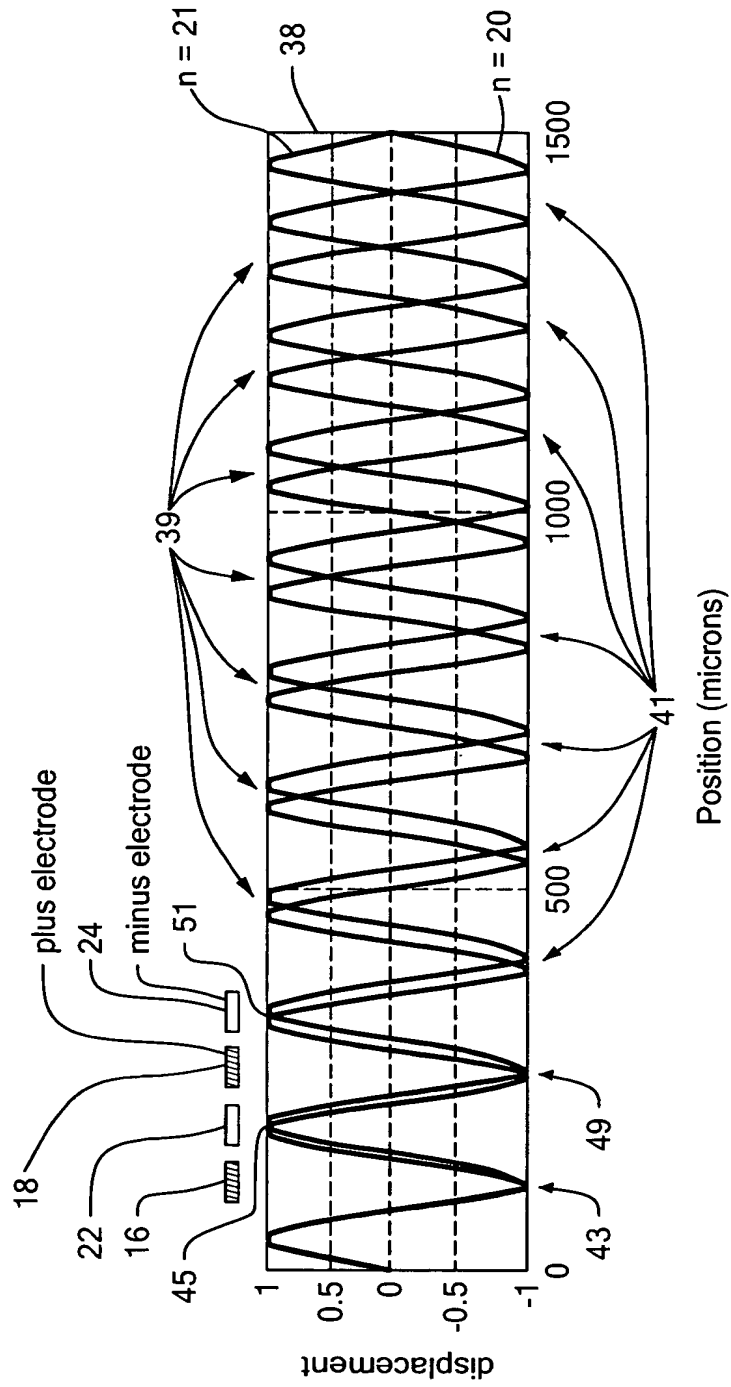
Applicants: Weinberg et al.  
 Title: FLEXURAL PLATE WAVE SENSOR  
 Serial No: 10/675,398  
 Docket No.: DR-312J  
 Atty: Roy J. Coleman Reg. No.: 48,863  
 1 of 22

1/22

10

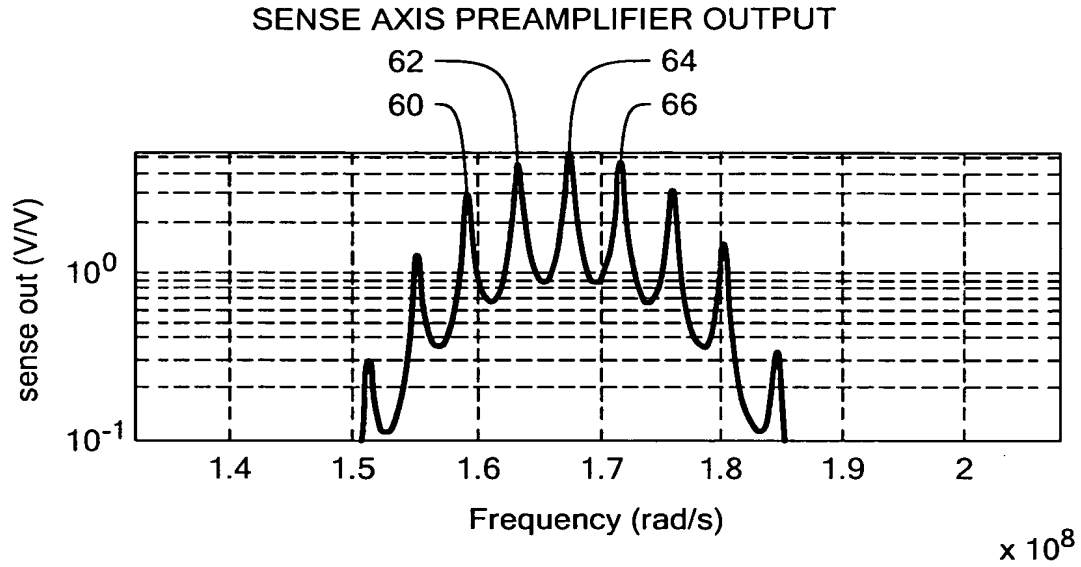


2/22

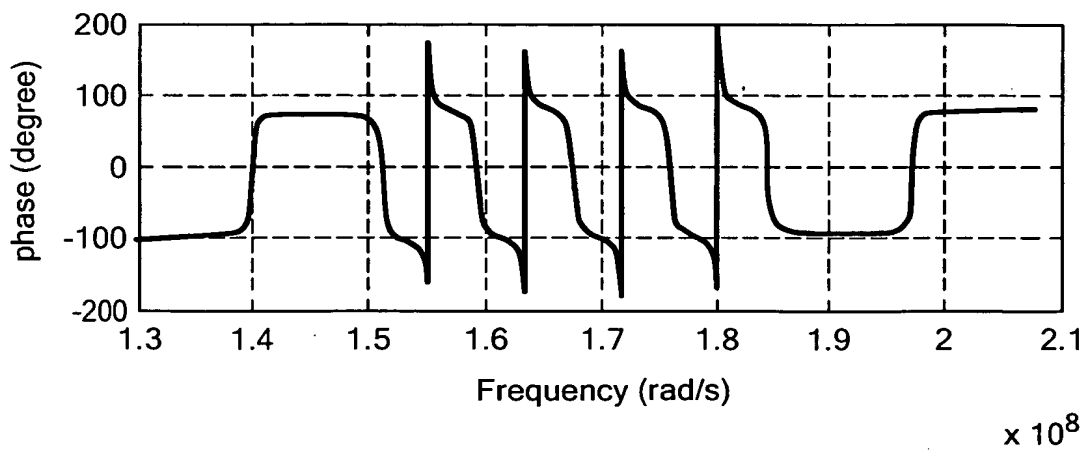


**FIG. 2**  
 PRIOR ART

3/22

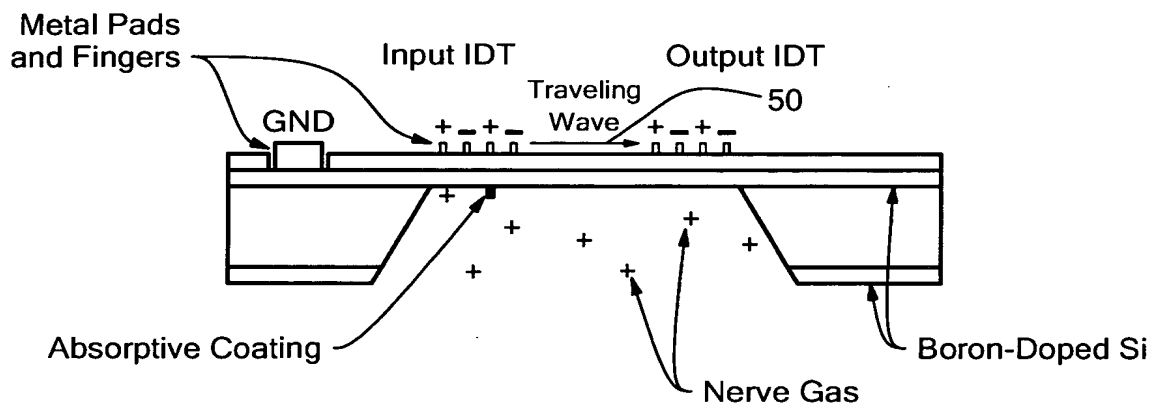


**FIG. 3A**



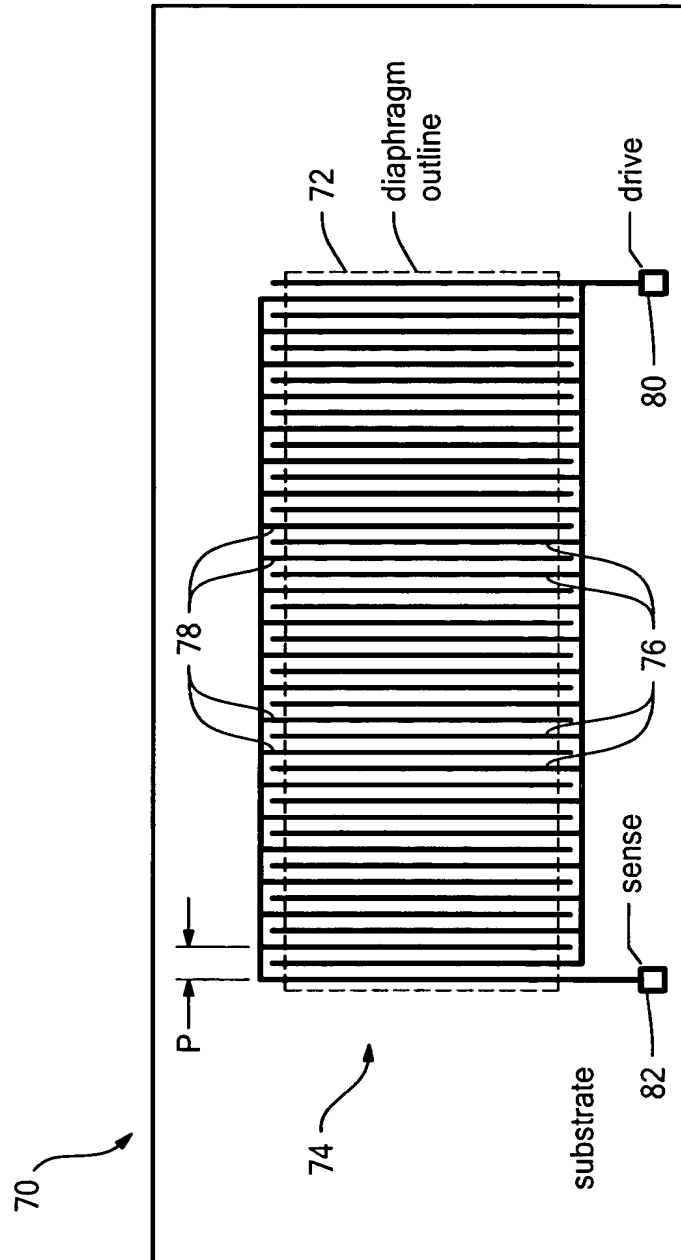
**FIG. 3B**

4/22



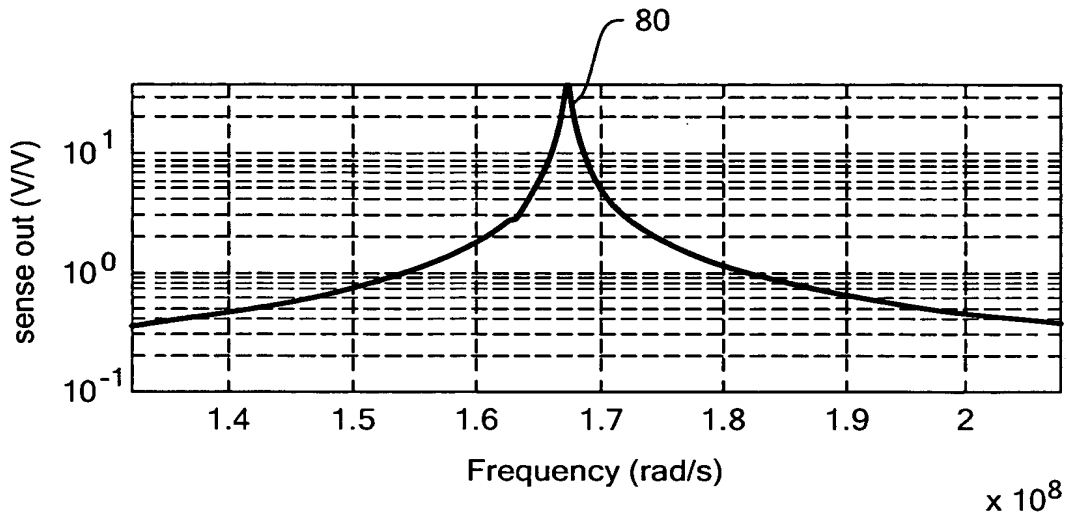
**FIG. 4**  
PRIOR ART

5/22

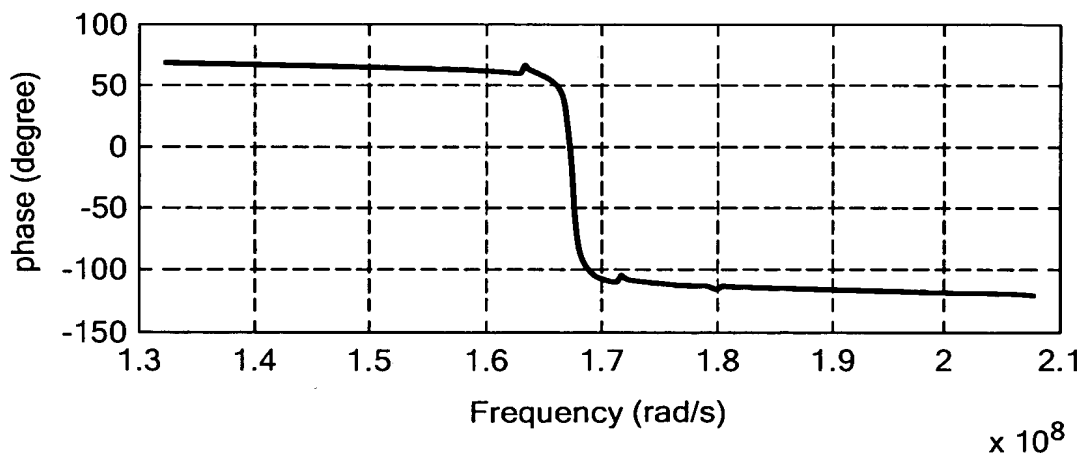


**FIG. 5**

6/22

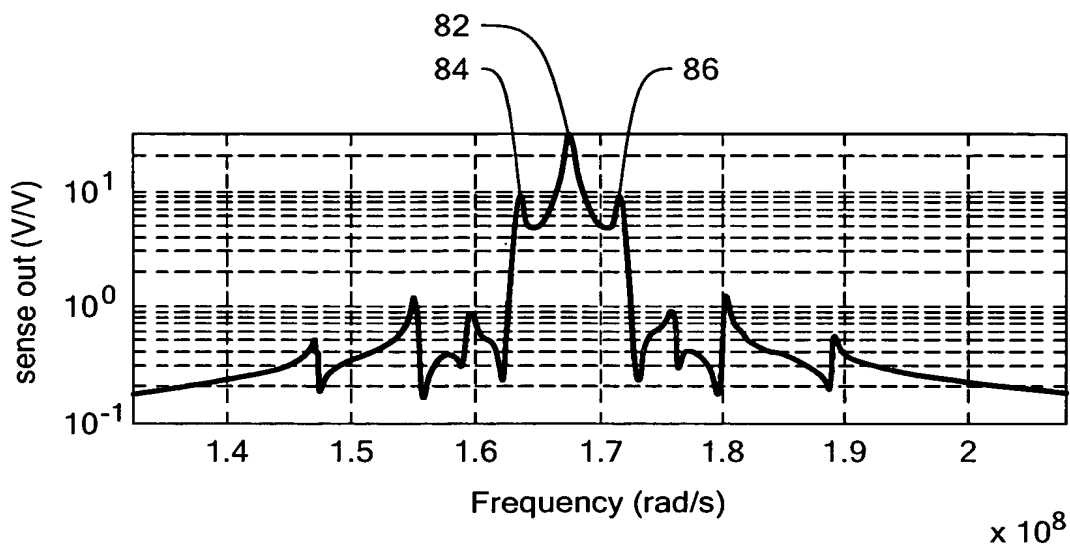


**FIG. 6A**

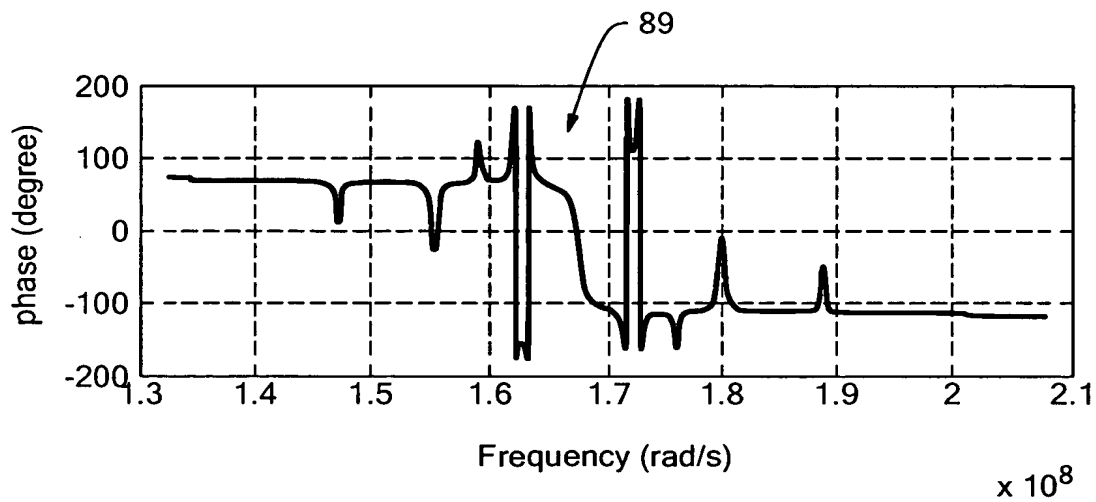


**FIG. 6B**

7/22

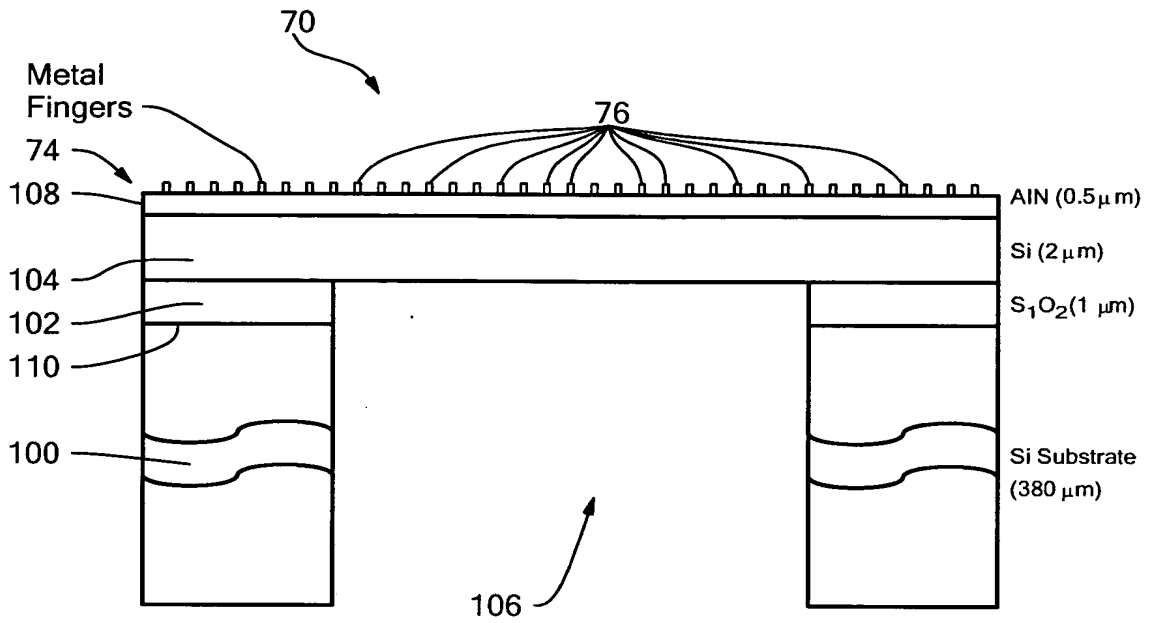


**FIG. 7A**

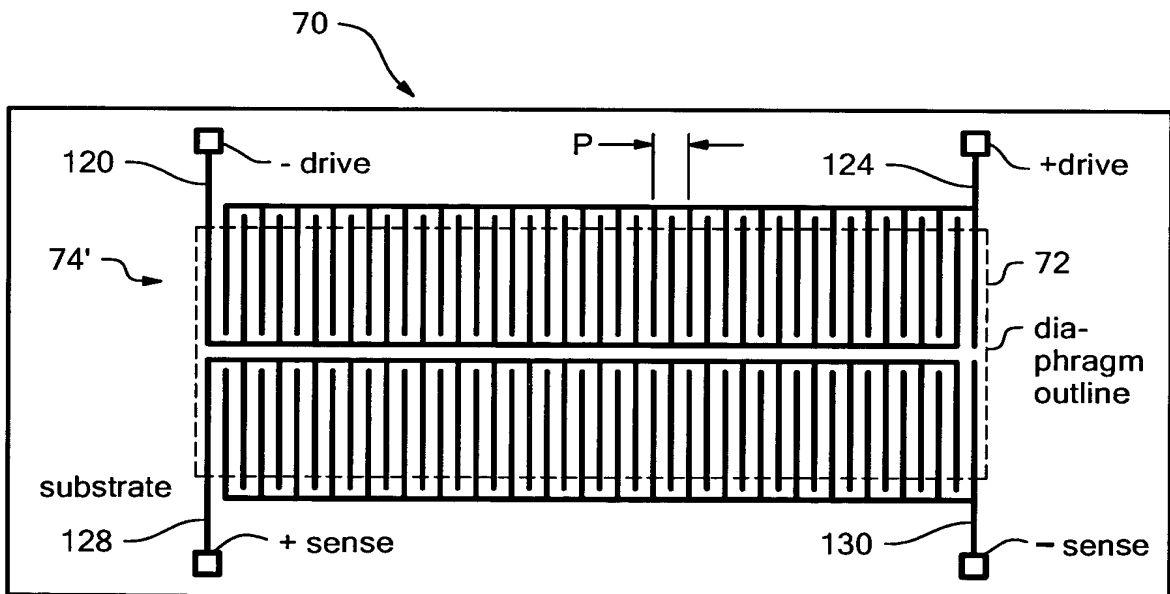


**FIG. 7B**

8/22



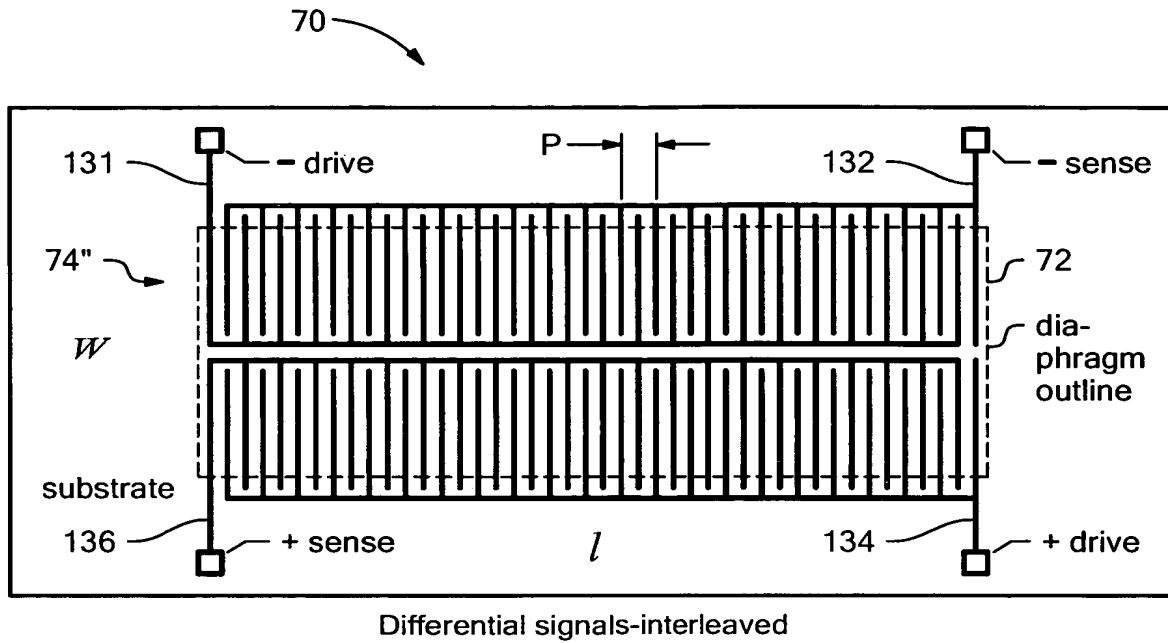
**FIG. 8**



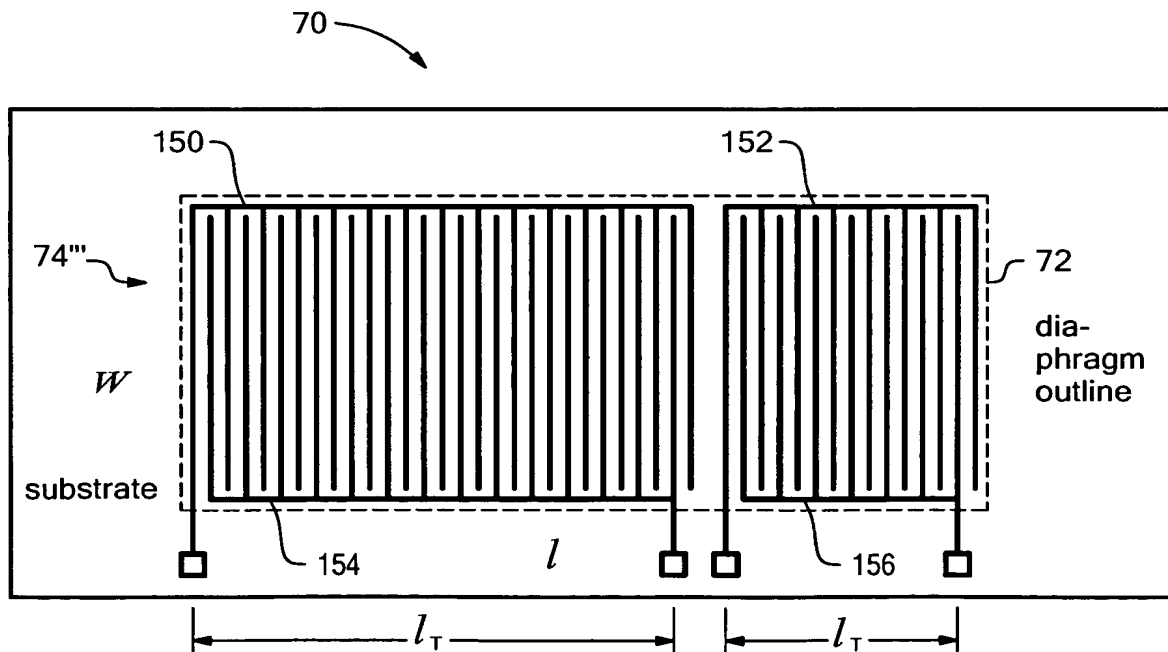
**FIG. 9**



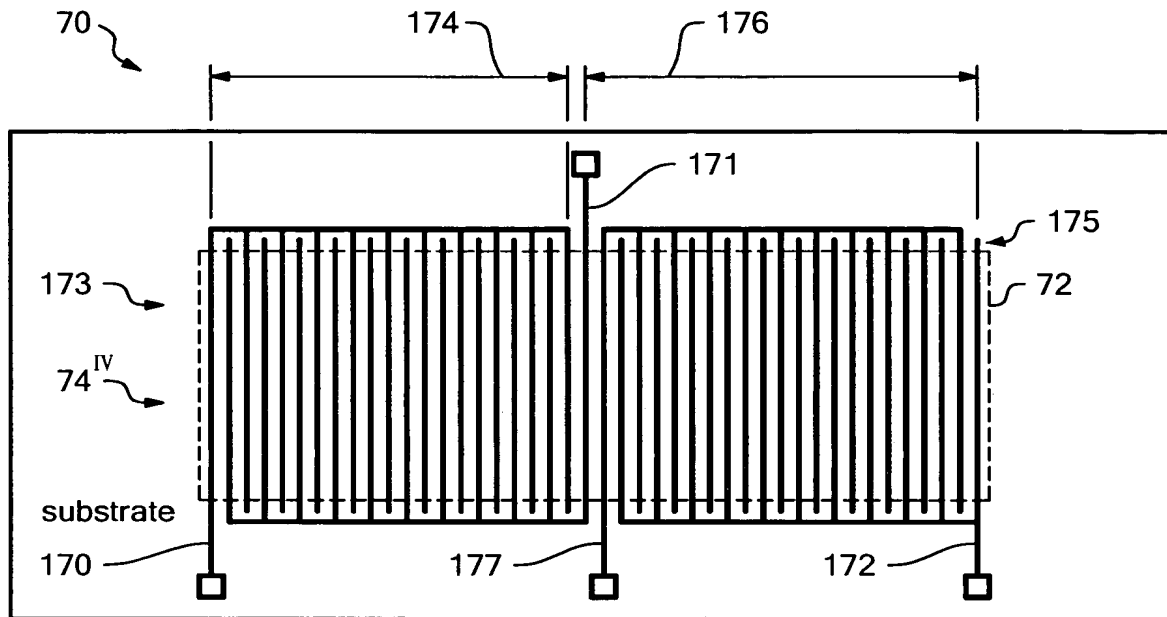
9/22



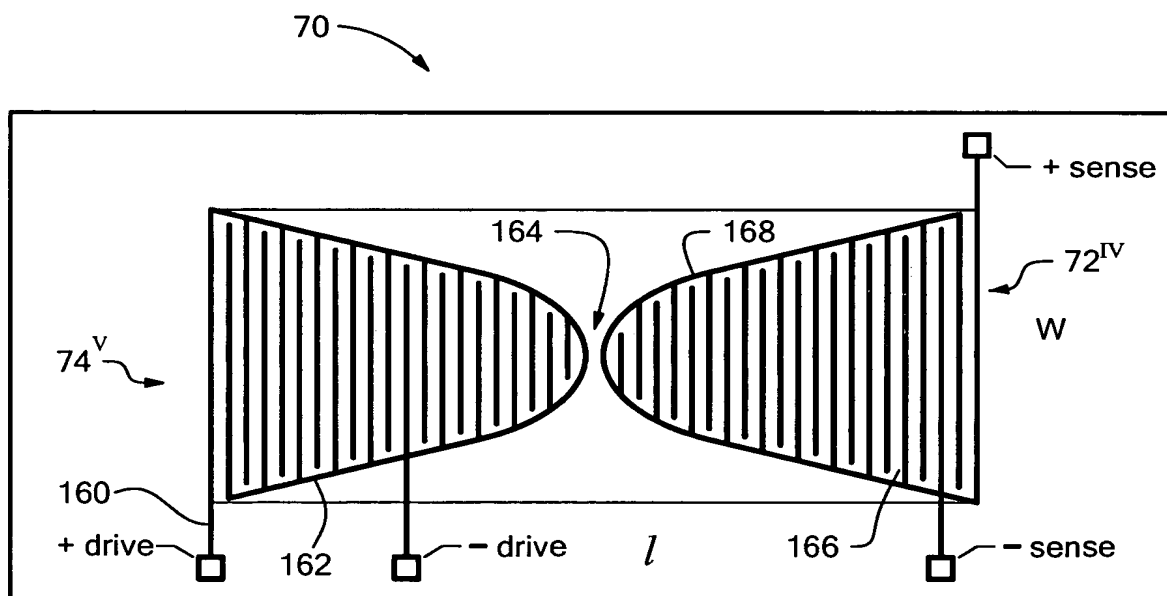
**FIG. 10**



**FIG. 11A**

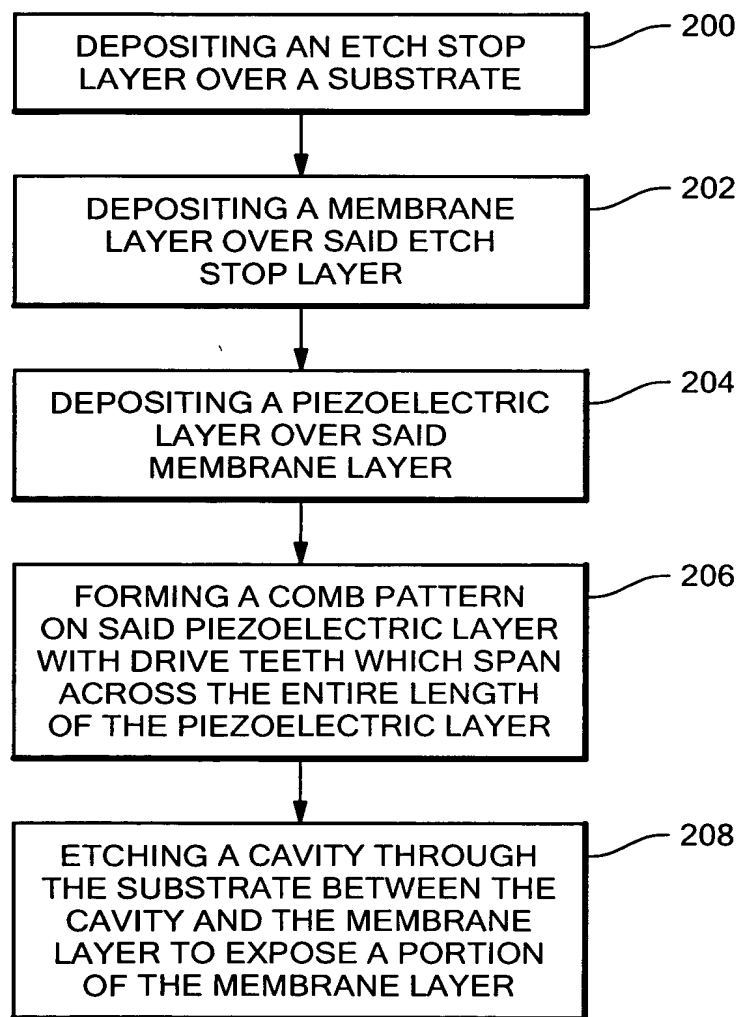


**FIG. 11B**



**FIG. 12**

11/22



**FIG. 13**

12/22

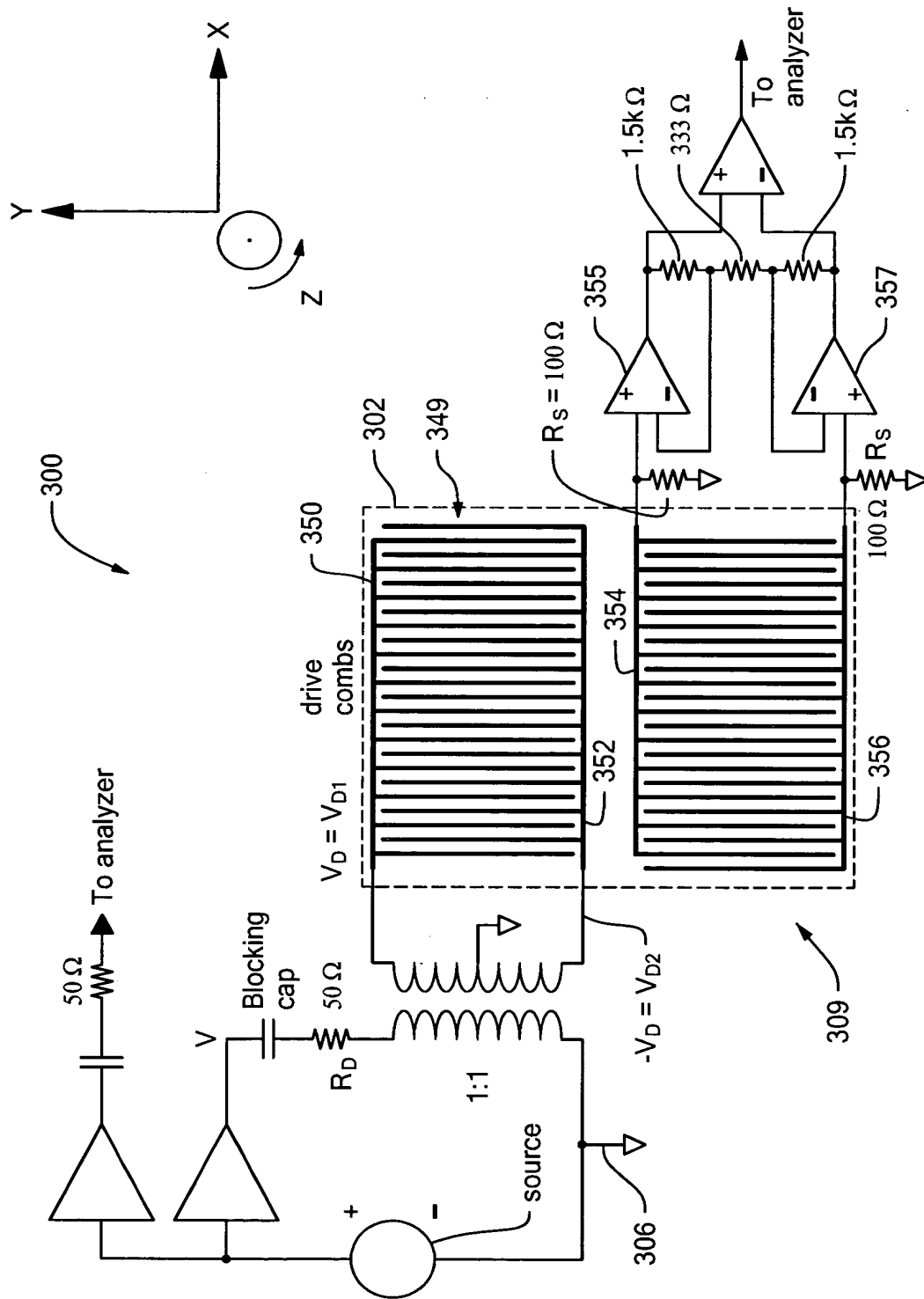
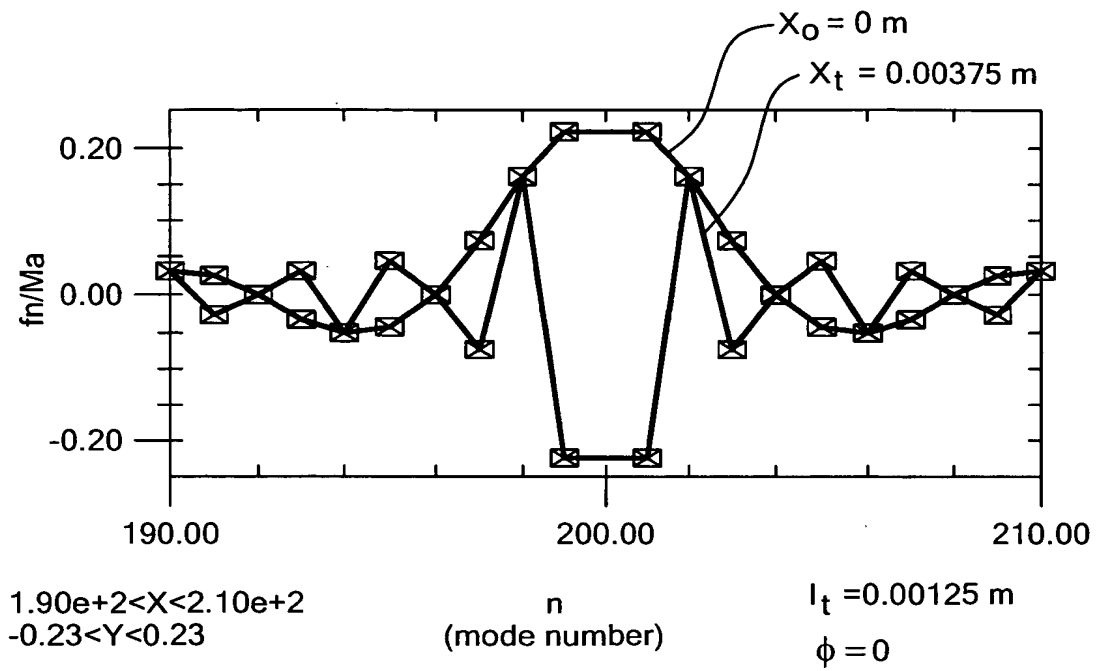
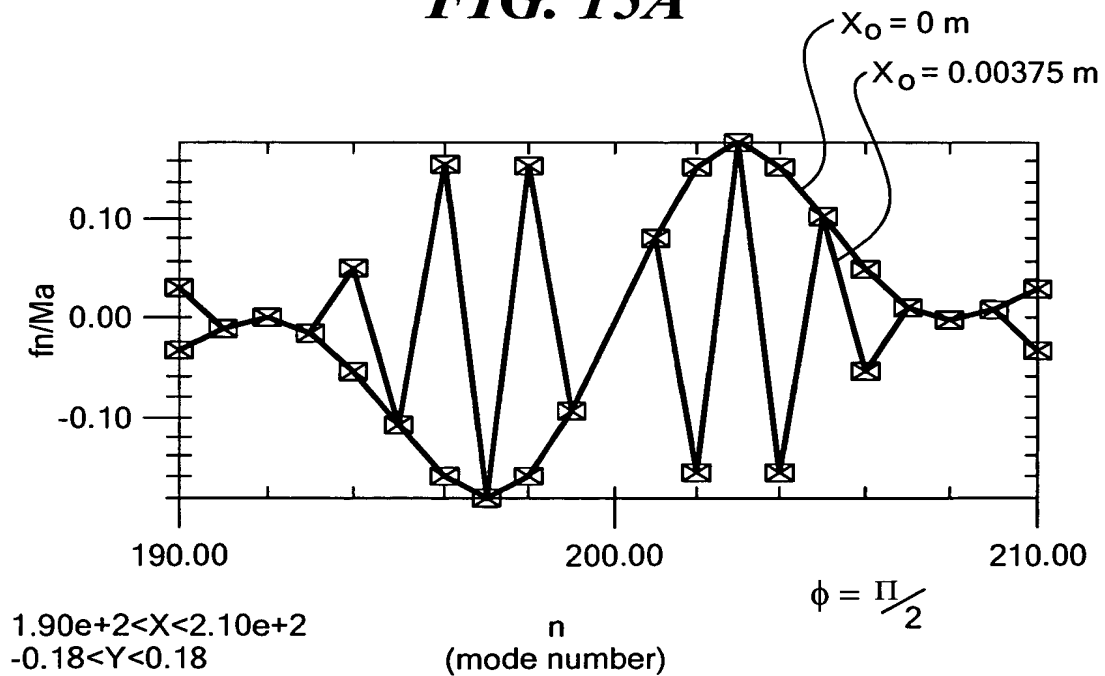


FIG. 14

13/22

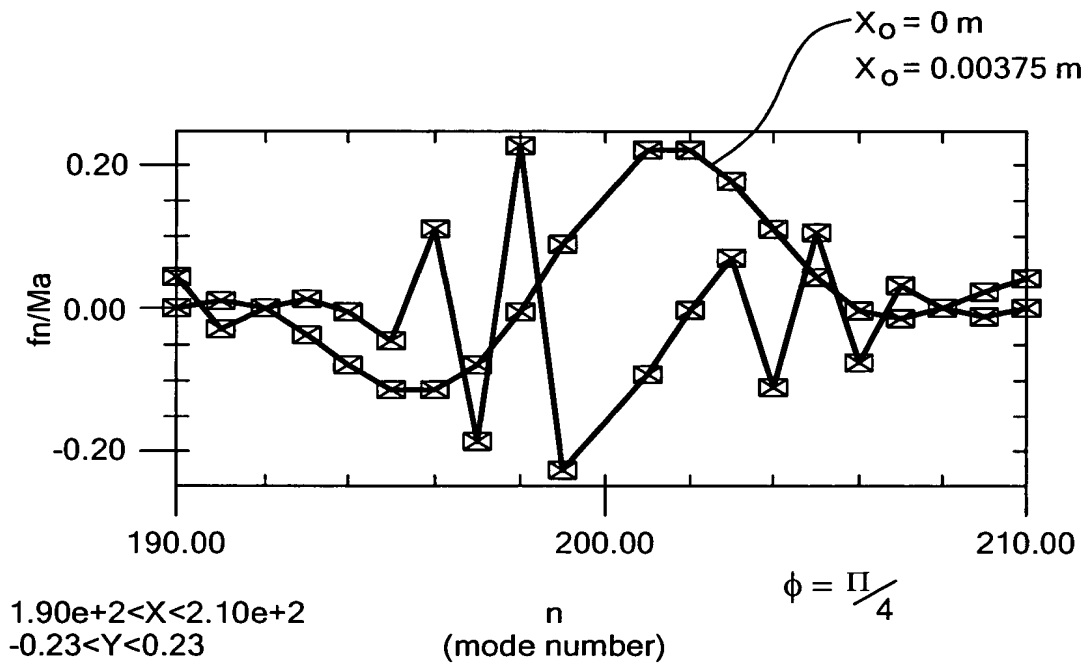


**FIG. 15A**



**FIG. 15B**

14/22



**FIG. 15C**

15/22

```

1: %MATLAB CODE FOR FPW CHEMICAL SENSOR MODAL FREQUENCY RESPONSE
2: %
3: %VARMODEEIG.M PARAMETERS FOR MICROCANARY
4: %PLATE WAVE RESONATOR DYNAMIC MODEL
5: %BASED ON FPW5 EXPANDED TO ARBITRARY NUMBER OF MODES BY ELI WEINBERG 7/19/99
6: %SEPTEMBER, 1999. CLOSER SCALING AND DIFFERENTIAL SENSE READING ADDED BY MSW.
7: %SEPARATE DRIVE AND SENSE INPUTS
8: %FIRST CODED 10/13/97
9: % 9/20/02 Calculate eigenvalues (photosensitivity investigation) difference from mechanical only
10: %LATEST RUN 9/20/02
11:
12: % 11/13/01 ATTEMPT THE CHEMICAL SENSOR Q = 100 CASES
13: clear, format compact, format short e, i=sqrt (-1);
14: diary c:\matlab6pl\mw\varmodec.dia
15: rd = 50/4 %xxx
16: %EXTERNAL DRIVE RESISTANCE. THE FACTOR OF FOUR ACCOUNTS FOR THE N=2
17: %TRANSFORMERS AND THE FACT THAT CD IS FOR +/- DRIVE IN PARALLEL.
18: rs = 100/2 %xxx %ONE HALF EXTERNAL SENSE RESISTANCE (ohm)
19: %THE FACTOR OF TWO ACCOUNTS FOR CS IS +/- DRIVE IN PARALLEL.
20: %GAINS BEFORE SOURCE VOLTAGE IS APPLIED TO FPW
21: gamp=1 %INPUT AMPLIFIER-BOTH INPUT AND REFERENCE LEGS HAVE SAME GAIN
22: ginst = 190 %INSTRUMENTATION AMPLIFIER GAIN-
23: %Completely differential amplifier with 10X second stage
24: gtran= 0.5 %TRANSFORMER GAIN
25: %VALUES FOR ALM = 0.5 MICRON, SI = TWO MICRONS
26: mp=2.47e-6 %xxx MASS PER UNIT LENGTH (KG/M)
27: dd = 8.781e-11 %STRUCTURAL RIGIDITY (N/M^2)
28: %q = 400 when bb = 1.033 %xxx %QUALITY FACTOR
29: bb = 1.033 %xxx damping (N-s/m^2)
30: 1 = 0.0015 %LENGTH OF DIAPHRAGM (m)
31: 1d = 3.750001E-5*19 %LENGTH OF THE DRIVE TRANSDUCER
32: 1s = 3.750001E-5*19 %LENGTH OF THE SENSE TRANSDUCER
33: md=38 %NUMBER OF HALF PERIODS IN TRANSDUCER
34: ms=38 %NUMBER OF HALF PERIODS IN TRANSDUCER
35: pd = 2*1d/md %PITCH OF DRIVE FINGERS (M)

```

**FIG. 16A**

16/22

```

36: ps = 2*ls/ms          %PITCH OF SENSE FINGERS (M)
37: %CALCULATE THE MODAL FORCING FUNCTION
38: phi= 0                %PHASE OF EIGENFREQUENCY (0 FOR PINNED, PI/4 FOR BUILT-IN)
39: thetad= 0             %PHASE OF TRANSDUCER (radians)
40: thetas = 0
41: xq=0                  %STARTING POSITION FOR FORCING COMBS
42: xs=1-ls-xd            %STARTING POSITION OF TRANSDUCER 20
43: %BEWARE CHANGING XD TO ADJUST TOLERANCES.
44: %SCALE FACTOR FOR LENGTH OF SENDER OR RECEIVER COMB
45: %MODEINT IS MSW FUNCTION BASED ON MACSYMA INTEGRATION
46:
47: nmode=input('Enter number of modes ')
48: model=input('Enter number of first mode in model ')
49:
50: for c0=1:nmode
51:     n(c0)=(model-1+c0);
52: end %MODE NUMBER, ROUGHLY NUMBER OF HALF WAVE LENGTHS
53:
54:
55:
56: alph = 0.004874/1.676e5^2
57: %piezo coupling coefficient for 100% transducer length (coul/m)
58: gamm = 9.39e-11*1.676e5^4 %piezo coupling coefficient (m/V)
59:
60: for cl = 1:nmode;
61:     pc(cl,1)=modeint(n(cl),md,1,ld,phi,thetad,xd);
62:     pc(cl,2)=modeint(n(cl),ms,1,ls,phi,thetas,xs);
63:     lam(cl)=n(cl)*pi/l; %eigenvalue of plate motion (1/m)
64:     gammad(cl)=gamm*pc(cl,1)/lam(cl)^4;
65:     gammas(cl)=gamm*pc(cl,2)/lam(cl)^4;
66:     alphad(cl)=alph*lam(cl)^2*pc(cl,1);
67:     alphas(cl)=alph*lam(cl)^2*pc(cl,2);
68:     wn(cl)=sqrt(dd/mp)*lam(cl)^2; %eigenfrequency (rad/s)
69:     damping(cl) = bb; %mode damping (N-s/m^2)
70:     k(cl) = mp*wn(cl)^2; %mode stiffness (N/m^2)
71:     checkalpha(cl) = 1-gammad(cl)*k(cl)*0.5*1/alphad(cl);

```

**FIG. 16B**



17/22

```

72: end
73: pc, wn, gammad, gammas, alphad, alphas, checkalpha
74:
75: cs = 7.567e-11*ls/1 %CAPACITANCE FROM SENSE COMBS TO GROUND, 2 POLES (F)
76: %EQUAL TO THAT CALCULATED IN TABLE 1. STRAYS MAY ADD MORE.
77: cd = 7.567e-11*ld/1 %DRIVE CAPACITANCE (F)
78: rfd=rs*ginst %SENSE AXIS RESISTOR AND INSTRUMENTATION AMPLIFIER
79:
80: %ENTER THE COEFFICIENTS OF THE DERIVATIVES
81: %STATES ARE [QD,QS,V1,X1,V2,X2,V3,X3]
82: ml=zeros(2+2*nmode, 2+2*nmode);
83: mra=zeros(2+2*nmode, 2+2*nmode);
84: ml(1,1)=(cd*rd);
85: ml(2,2)=cs*rs;
86: mra(1,1)=-1;
87: mra(2,2)=-1;
88:
89: for c2=1:nmode
90:     ml(2*c2+1, 2*c2+1)=mp;
91:     ml(2*c2+2, 2*c2+1)=1;
92:     ml(2*c2+1, 1)=rd*k(c2)*gammad(c2);
93:     ml(2*c2+1, 2)=rs*k(c2)*gammas(c2);
94:
95:     mra(2*c2+1, 2*c2+1)=-damping(c2);
96:     mra(2*c2+1, 2*c2+2)=-k(c2);
97:     mra(2*c2+2, 2*c2+1)=1;
98:     mra(1, 2*c2+2)=alphad(c2);
99:     mra(2, 2*c2+2)=alphas(c2);
100: end
101:
102: %ENTER COEFFICIENTS OF DRIVE VOLTAGE
103: mrb=zeros(2*nmode+2, 1);
104: mrb(1)=cd;
105: for c3=1:nmode
106:     mrb(2*c3+1)=k(c3)*gammad(c3);
107: end

```

**FIG. 16C**

18/22

```

108:
109: %ml, mra, mrb
110:
111: %SET UP THE STATE MATRICES
112: invml=inv(ml);
113: checkinvml=invml*ml
114: a=invml*mra
115: b=invml*mrb
116: [evec,eval]=eig(a);
117: damp(eval)
118: %eval
119: %evec
120:
121: %PICK OUT THE DIFFERENCE IN THE EIGENFREQUENCIES
122: yy=sort(damp(eval)); %PLACE THE EIGENVALUES IN ORDER. CHECK THAT THE POLES CORRESPONDING TO THE
123: %RC ARE HIGHER THAN THE MECHANICAL POLES
124: for ii = 1:nmode
125:     wnc1(ii)=yy(2*ii-1); %MAKE THE VECTORS OF DIFFERENT LENGTHS SIMILAR AND OMIT THE TWO FASTEST POLES
126: end
127: wndiff=wnc1-wn %SHIFT FROM MECHANICAL RESONANCES TO CLOSED LOOP
128:
129: %OUTPUTS ARE AMPLITUDE, SENSE PREAMPLIFIER OUTPUT, AND INPUT CHARGE THROUGH RD
130: %WITH GIN = 1 INPUT IS INPUT TO 10X AMPLIFIER WHICH IS MEASURED BY
131: %ANALYZER
132: gin = 1 %SOURCE TO PREAMP INPUT
133: c=zeros(3, 2+2*nmode);
134: for c4=1:nmode
135:     c(1, 2*c4+2)=1;
136:     c(2, :)=a(2, :)*rfb;
137:     c(3, 1)=.5;
138: end
139: c=c*gamp*gtran*gin
140: d=[0;b(2,:)*rfb; 0]*gamp*gtran*gin
141: %w=logspace(7,9,200);

```

**FIG. 16D**

19/22

# FIG. 16E

```

142: nw=1001
143: dw=(1.1*wn(nmode)-0.9*wn(1))/(nw-1);
144: w=[0.9*wn(1):dw:1.1*wn(nmode)];
145: wmax=w(length(w))
146: [m1,p1]=unbode(a,b,c,d,1,w);
147: xmax = max(m1(:,1))
148: vmax = max(m1(:,2))
149: figure(1),clf,subplot(2,1,1)
150: semilogy(w,m1(:,2)),grid,xlabel('frequency (rad/s)')
151: ylabel('sense out (V/V)'),axis([w(1),wmax,0.01,max(m1(:,2))])
152: title('SENSE AXIS PREAMPLIFIER OUTPUT')
153: subplot(2,1,2)
154: plot(w,p1(:,2)),grid,xlabel('frequency (rad/s)'), ylabel('phase (deg)')
155: figure(2),clf,subplot(2,1,1)
156: semilogy(w,m1(:,1)),grid,xlabel('frequency (rad/s)')
157: ylabel('amplitude (m/V)'),axis([w(1),wmax,0.01*max(m1(:,1),max(m1(:,1)))])
158: title('MOTION AMPLITUDE')
159: subplot(2,1,2)
160: plot(w,p1(:,1)),grid,xlabel('frequency (rad/s)'), ylabel('phase (deg)')
161: figure(3),clf,subplot(2,1,1)
162: semilogy(w,m1(:,3)),grid,xlabel('frequency (rad/s)')
163: ylabel('charge (C/V)')
164: title('DRIVE CHARGE')
165: subplot(2,1,2)
166: plot(w,p1(:,3)),grid,xlabel('frequency (rad/s)'), ylabel('phase (deg)')
167: % [z,p,k]=ss2zp(a,b,c,d,1) %OBTAIN THE POLES AND ZEROS OF TRANSFER FUNCTION
168: diary off
169
170:

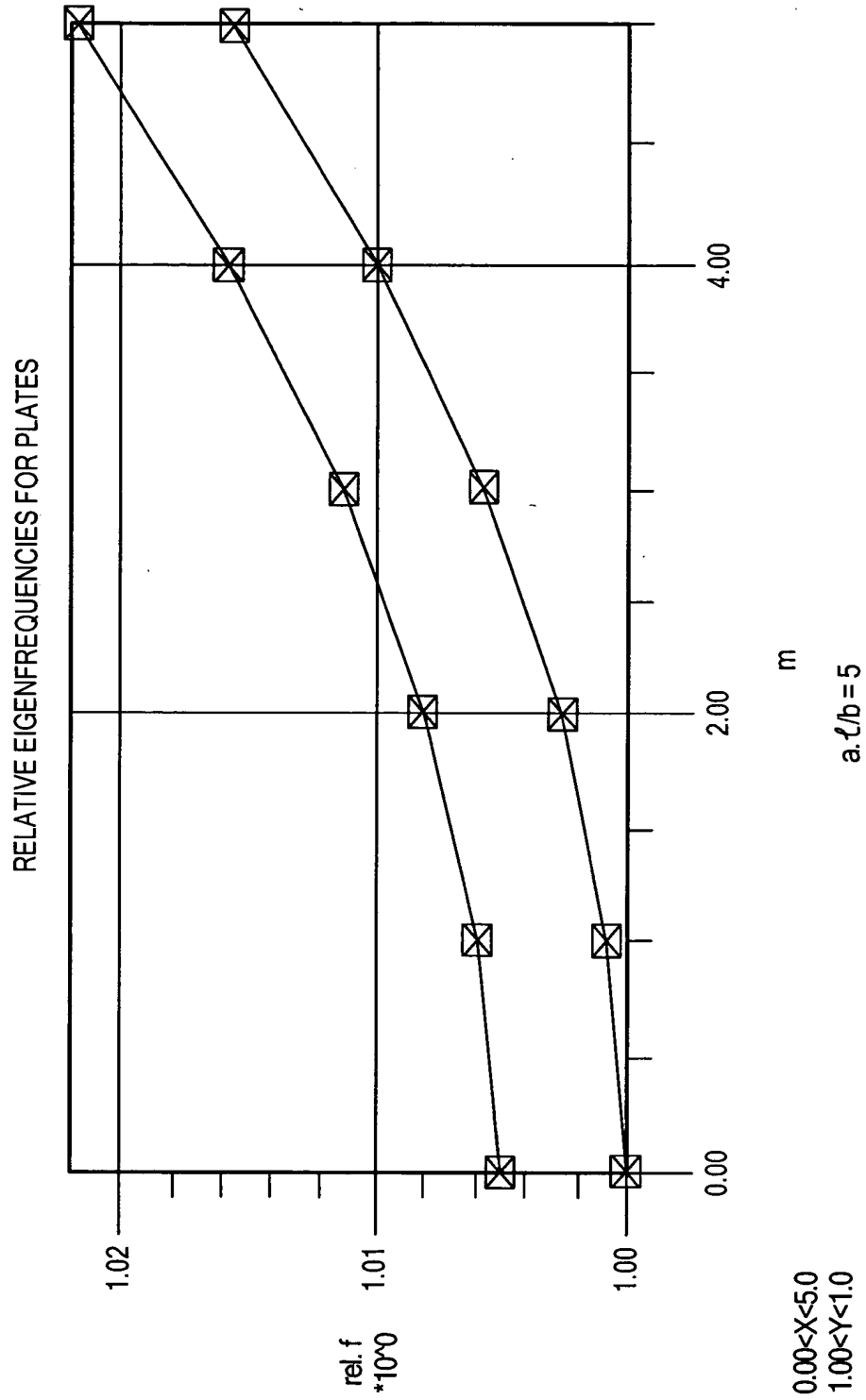
```

20/22

```
1: function [pc]=modeint(n,m,l,lt,phi,theta,xo);
2: %FUNCTION TO OBTAIN THE MODAL FORCING FUNCTION NORMALIZED TO ONE
3: %DERIVED BY MACSYMA AND CODED 10/20/97
4: pc = -lt*sin(((pi*lt*n+pi*lt*m)*xo-l*lt*theta-l*lt*phi+pi*lt^2*....
5:   *n+pi*lt*m)/(l*lt))/(pi*(lt*n+l*m))+lt*sin(((pi*lt*n+....
6:   pi*lt*m)*xo-l*lt*theta-l*lt*phi)/(l*lt))/(pi*(lt*n+l*m))+lt*....
7:   sin(((pi*lt*n-pi*lt*m)*xo+l*lt*theta-l*lt*phi+pi*lt^2*n-pi*....
8:   *lt*lt*m)/(l*lt))/(pi*(lt*n-l*m))-lt*sin(((pi*lt*n-pi*lt*m)*....
9:   xo+l*lt*theta-l*lt*phi)/(l*lt))/(pi*(lt*n-l*m));
10:
```

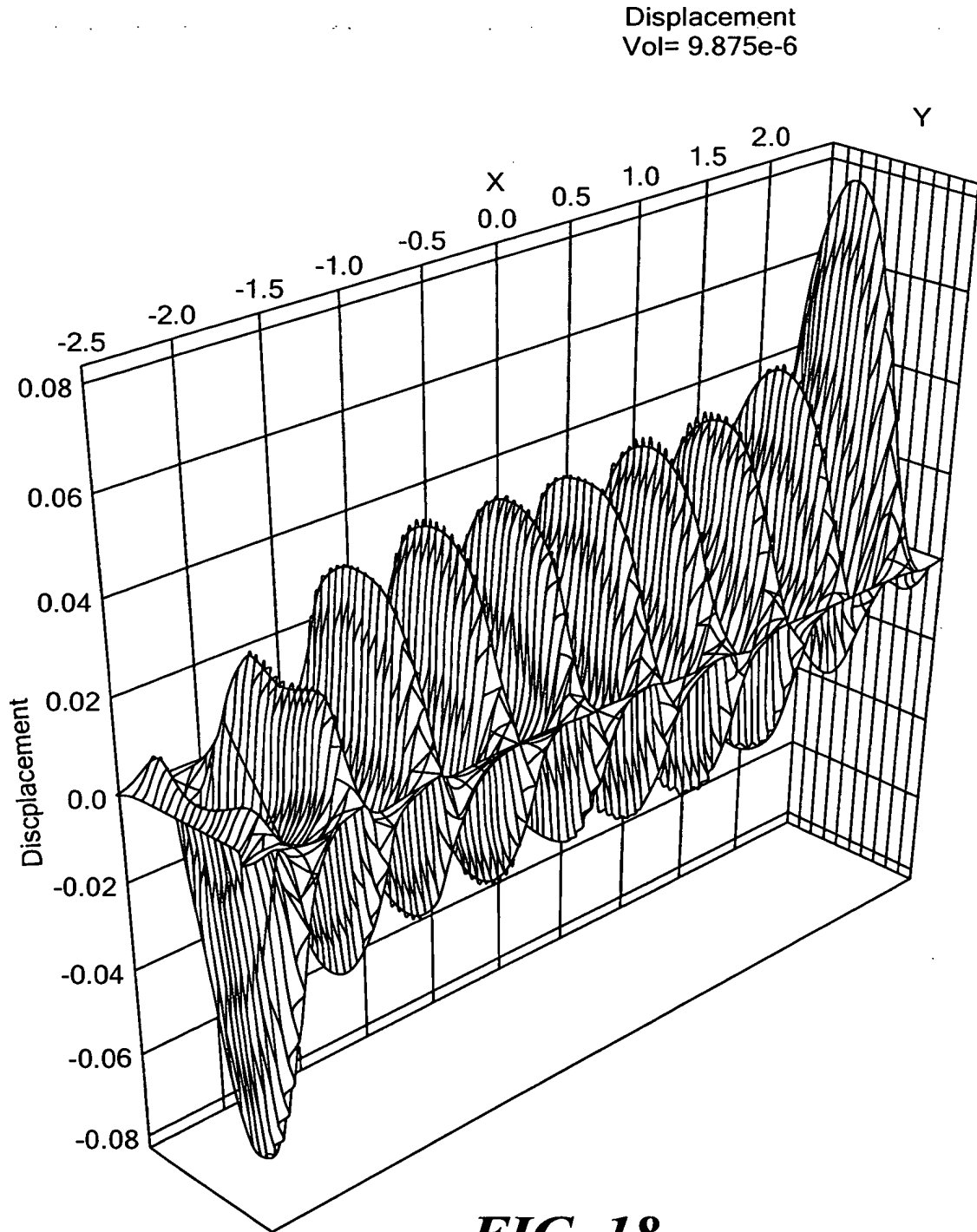
**FIG. 16F**

21/22



**FIG. 17**

22/22



**FIG. 18**

Static Plate Deflections  
for Sinusoidal Load